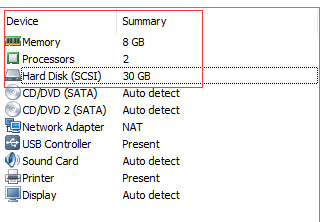
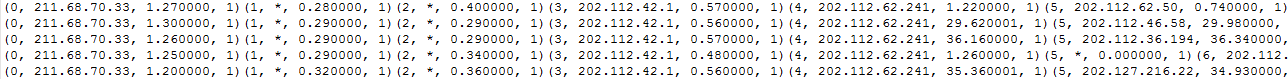
**实验的配置**

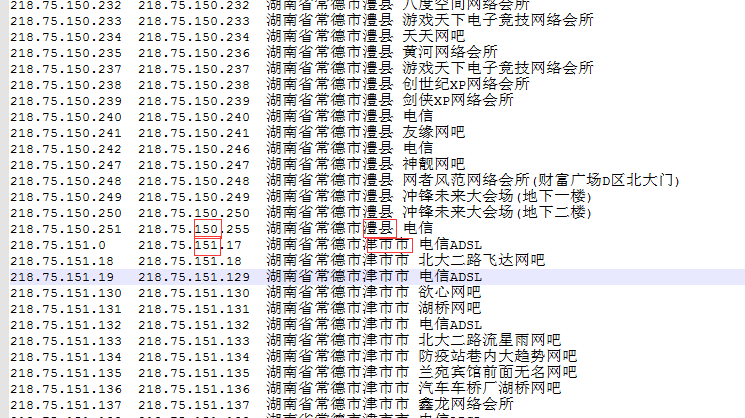


操作系统是ubuntu-kylin 64位，c++编程

**拓扑数据**

老师您给的两个拓扑数据”poppath”和” CTrace”,poppath只能提出一组拓扑数据，”CTrace”里的数据有15组，数据的形式是如下这样

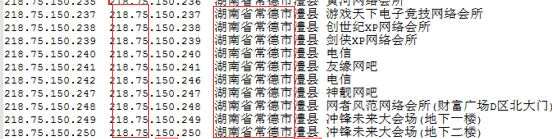
其中的IP地址实际的物理地址如下：



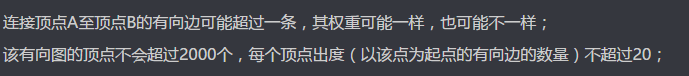
1）以整个IP地址抽象为一点，点数太多太多，算法运行时间太长。

2）取前缀抽象出点，比如长沙市到常德市就有**多条重边**，再随机风险共享链路组，我们的算法不能出现trap(即找到第一条路径AP,找不到第二条路径BP，实际是存在AP和BP分离路径对)的情形。R如果把所有重边看成一条边，则出现新的问题，比如长沙到常德就只能有一条路径，能肯定不存在分离路径。

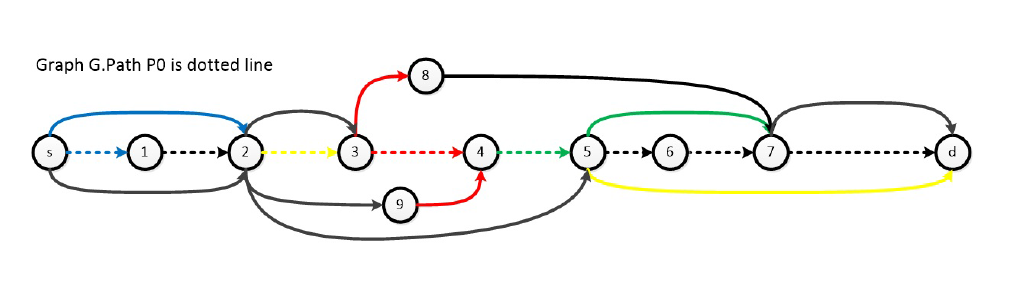
3）再取更前缀抽象出点，点数就太少了。



我主要用从华为比赛得到的数据（华为曾今说他们比赛的数据是从实际网络抽象出来的）



他们的数据保证源节点到目的节点在没有SRLG的情况下，一定存在分离路径。图是稀疏图（出度20），随机设置SRLG，还是得不出我们算法出现trap的情形，我再把论文上的那个拓扑结构拼在一起，在自己设置SRLG

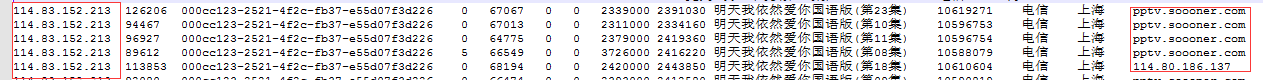


论文上的那个拓扑结构

才会出现trap的情形.

**源节点和目的节点**

源节点和目的节点是从老师您给的数据”12\_video\_info”中提取的，



数据的内容说明如下

每行一条记录，用tab分开，各个列的含义如下（从第一到最后一列）：

用户IP

开始时间(相对时间，从0开始）

用户ID

拖动次数

最后码流率

码流切换次数

用户抽样速率（字节数）

观看时长（毫秒）

视频长度（毫秒 ）

视频名称

频道ID

运营商

省份

服务器IP

我用用户IP和服务器IP哈希成两个32位整数，如果这两个整数大于测试的拓扑图的实际点数，则取模。比如源节点是123542，目的节点是236442，而拓扑图点事为1111，则源节点为123542%1111，目的节点为236442%1111。